SpyWolf

Project Audit

Project:

Contexia

April 21, 2022

Official Document

SpyWolf.co

Overview

This audit has been prepared for **Contexia** to review the main aspects of the project to help investors make an informative decision in the research process.

You will find a a summarized review of the following main key points:

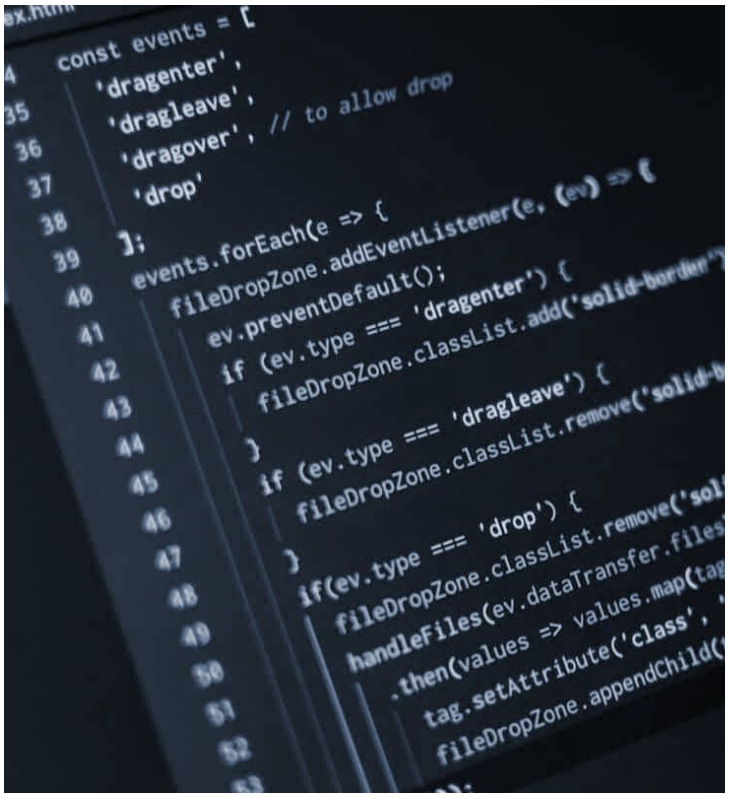
- Contract's source code
- Project and team
- Website
- Social media & online presence

NOTE: We ONLY consider a project safe if they receive our "Certificate of Trust" NFT. This report only points out any potential red flags found in our analysis. Always do your own research before investing in a project.

Smart Contract Review

The contract review process pays special attention to the following:

- Testing the smart contracts against both common and uncommon vulnerabilities
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.



"The results of this audit are purely based on the team's evaluation and does not guarantee nor reflect the projects outcome and goal"
- SpyWolf Team

Smart Contract Summary

Contract Name	ContexiaToken
Ticker	CON
Contract	0x1eD959d43696a045f7cF59748f04CE272922917F
Network	Avalanche
Language	Solidity
Tax	**Buy: 0%** **Sell: 30%**
Total Supply	2,000,000
Status	Not launched

Current stats

Burn	No burnt tokens
LP Address	Liquidity not added yet
MaxSellAmount	50,000

Issues Checking Status	
Design Logic	**Passed ✓**
Compiler warnings.	**Passed ✓**
Private user data leaks	**Passed ✓**
Timestamp dependence	**Passed ✓**
Integer Overflow and Underflow	**Passed ✓**
Race conditions and Reentrancy. Cross-function race conditions	**Passed ✓**
Possible delays in data delivery	**Passed ✓**
Oracle calls	**Passed ✓**
Front running	**Passed ✓**
DoS with Revert	**Passed ✓**
DoS with block gas limit	**Passed ✓**
Methods execution permissions	**Passed ✓**
Economy model	**Passed ✓**
The impact of the exchange rate on the logic	**Passed ✓**
Malicious Event log	**Passed ✓**
Scoping and Declarations	**Passed ✓**
Uninitialized storage pointers	**Passed ✓**
Arithmetic accuracy	**Passed ✓**
Cross-function race conditions	**Passed ✓**
Safe Zeppelin module	**Passed ✓**
Fallback function security	**Passed ✓**

Featured Wallets

Owner address	0x0C61a18a7AcAe5304D10b9eEC796a91dDc4CE719
*Reward wallet	0x0000000000000000000000000000000000000000
*Treasury wallet	0x0000000000000000000000000000000000000000
LP token address	Liquidity not added yet

Top 3 Unlocked Wallets

Wallet 1 (100%)	Same as owner

Tokens are not distributed yet

***This wallet can be changed by owner**

Security Threads

Owner can change sell fees up to 30% and transfer fees up to 100%. Owner can’t set buy fees, but all buys will be charged as transfers until the LP address is set via the setPair function.

```
function setPair(address _pair) public onlyOwner returns (bool) {
    require(pair == address(0), "already set");
    pair = _pair;
    return true;
}

function setBuyAndSellFees(
    uint32 _transferFee,
    uint32 _sellFee,
    uint32 _feeDenominator
) public onlyOwner returns (bool) {
    transferFee = _transferFee;
    sellFee = _sellFee;
    feeDenominator = _feeDenominator;
    require(
        sellFee <= (feeDenominator * 30) / 100,
        "Can't be more than 30%"
    );
    return true;
}
```

Security Threads

There is function to change max sell limit but owner can’t use it, because the required amount is higher than the scope of uint32. Current max sell transaction limit is 50 000 tokens.

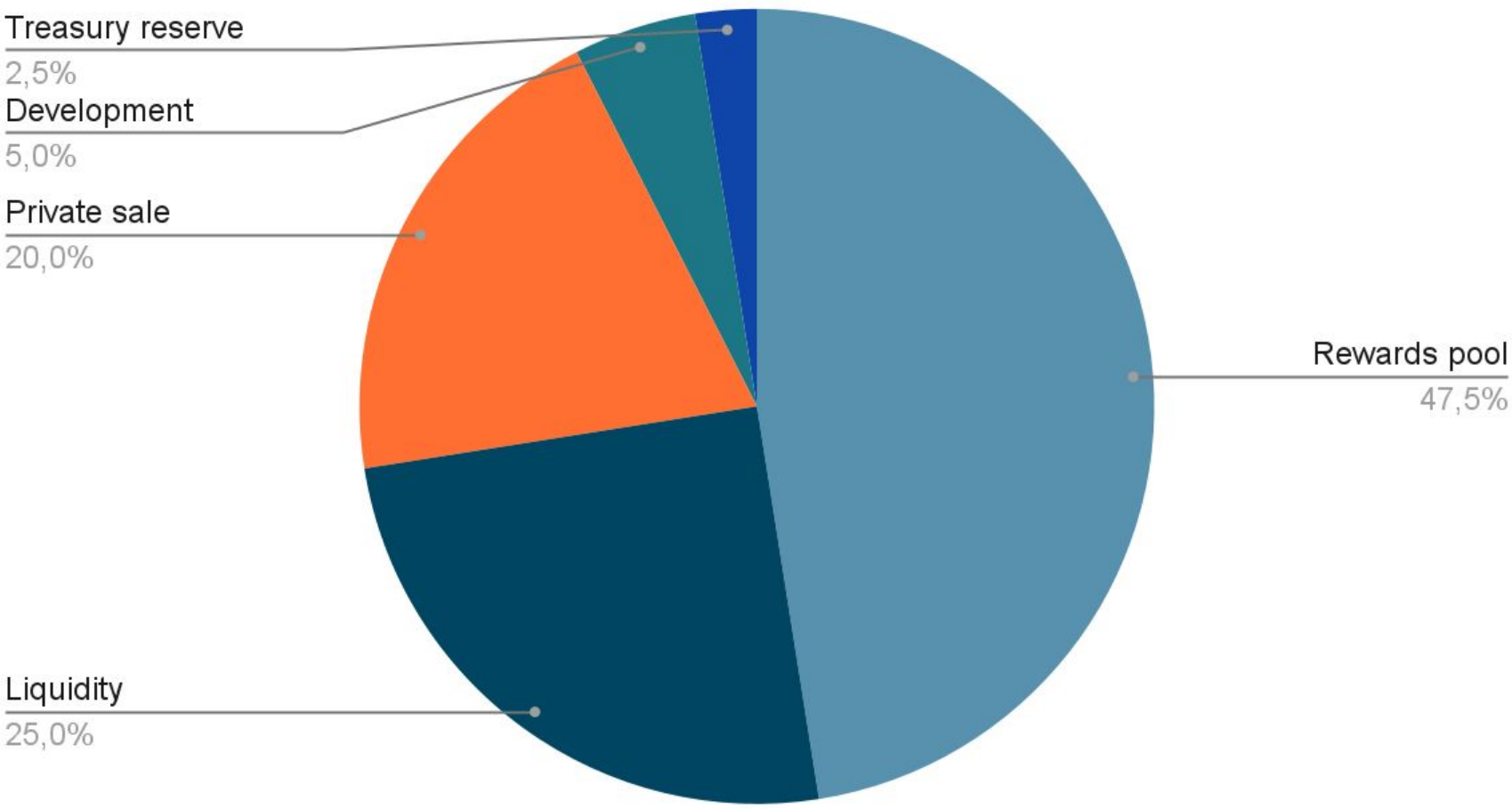
```
uint256 public maxSellAmount = 50_000e18;

function setMaxSellLimit(uint32 _amount)
    public
    onlyOwner
    returns (bool)
{
    require(
        _amount >= 1_000e18,
        "Can't be more than 30%"
    );
    maxSellAmount = _amount;
    return true;
}
```

Tokenomics

According to project's whitepaper

Total supply 2,000,000 **CON**

- 47.5% - Rewards pool
- 25% - Liquidity
- 20% - Private sale
- 5% - Development
- 2.5% - Treasury reserve



Smart Contract Summary

Contract Name	**ContexiaNodePurchaser**
Ticker	N/A
Contract	0x8850D7CE86f13088CCBD0Ba8C63a7C3cfFf93CbF
Network	Avalanche
Language	Solidity
Tax	**CashoutTreasuryFee: 2%** **CashoutPoolFee: 8%**
Total Supply	N/A
Status	Not launched

Featured Wallets

Owner address	0x0C61a18a7AcAe5304D10b9eEC796a91dDc4CE719
*Reward pool	0x509f74eA9806Adf7fC03d1B0D05D9B56Baa96e7C
*Treasury	0x62Ac812366297ab454dD4442a6b3D00f23590373

***This address can be changed by the owner**

Security Threads

Owner can blacklist address to create nodes and withdraw rewards.

```
function blacklistMalicious(address account, bool value)
    external
    onlyOwner
{
    _isBlacklisted[account] = value;
}
```

Owner can change treasury and reward pool wallets.

```
function updateTreasury(address _addr) external onlyOwner {
    treasury = _addr;
}
function updateRewardsWall(address payable _addr) external onlyOwner {
    rewardPool = _addr;
}
```

Owner can change fees up to 100%

```
function updateFees(
    uint256 _rewardPoolFee,
    uint256 _treasuryFee,
    uint256 _claimRewardPoolFee,
    uint256 _claimTreasuryFee
) external onlyOwner {
    rewardPoolFee = _rewardPoolFee;
    treasuryFee = _treasuryFee;
    cashoutPoolFee = _claimRewardPoolFee;
    cashoutTreasuryFee = _claimTreasuryFee;
}
```

Security Threads

Owner can send funds from contract to any wallet.

```
function boostReward(address _account, uint256 amount) public onlyOwner {
    if (amount > address(this).balance) amount = address(this).balance;
    payable(_account).transfer(amount);
}
```

Owner can change each node price, rewards payout and rewards claim time.

```
function changeNodePrice(uint256[3] memory newNodePrice) public onlyOwner {
    nodes[0]._changeNodePrice(newNodePrice[0]);
    nodes[1]._changeNodePrice(newNodePrice[1]);
    nodes[2]._changeNodePrice(newNodePrice[2]);
}

function changeRewardPerNode(uint256[3] memory newPrice) public onlyOwner {
    nodes[0]._changeRewardPerNode(newPrice[0]);
    nodes[1]._changeRewardPerNode(newPrice[1]);
    nodes[2]._changeRewardPerNode(newPrice[2]);
}

function changeClaimTime(uint256[3] memory newTime) public onlyOwner {
    nodes[0]._changeClaimTime(newTime[0]);
    nodes[1]._changeClaimTime(newTime[1]);
    nodes[2]._changeClaimTime(newTime[2]);
}
```

Smart Contract Summary

Contract Name	ContexiaNode
Ticker	N/A
Contract	0x22CEf2d14035C88e4eBdc408F10a616faceEB42B
Network	Avalanche
Language	Solidity
Tax	**Claim tax: 13%**
Total Supply	N/A
Status	Not launched

Security Threads

Owner can change node's claim time period, day fees, maintenance and claim fees.

```
function _changeClaimTime(uint256 newTime) external isRole(OWNER_ROLE) {
    isolationPeriod = newTime;
}
function changeFeeDueDays(uint256 _days) external isRole(OWNER_ROLE) {
    feeDueDays = _days;
}
function updateClaimTax(uint256 _tax) external isRole(OWNER_ROLE) {
    claimTax = _tax;
}
function setNodeMaintenanceFees(uint256 _nodeMaintenanceFee)
    external
    isRole(OWNER_ROLE)
{
    nodeMaintenanceFee = _nodeMaintenanceFee;
}
function updatePriceFee(AggregatorV3Interface _feed)
    external
    isRole(OWNER_ROLE)
{
    priceFeed = _feed;
}
```

Owner can grant role to any address.

```
function grantRole(bytes32 role, address account)
 public virtual override onlyRole(getRoleAdmin(role)) {
    _grantRole(role, account);
}
```

Owner can change rewards payout for node.

```
function _changeRewardPerNode(uint256 newPrice)
    external
    isRole(OWNER_ROLE)
{
    rewardPerNode = newPrice;
}
```

Security Threads

Owner can change booster nodes limit and reward parameters of user's node.

```
function setPerNodeBoosterLimit(uint256 _limit)
    external
    isRole(OWNER_ROLE)
{
    perNodeBoosterLimit = _limit;
}

function updateProductionRate(
    address _user,
    uint256 _nodeIndex,
    uint256 _productionRatePer
) external isRole(NODE_BOOSTER_ROLE) {
    NodeEntity storage node = _nodesOfUser[_user][_nodeIndex];
    require(
        node.boosterInfo.winstar < perNodeBoosterLimit,
        "NODE: Already Boosted"
    );

    node.boosterInfo.winstar += 1;
    node.rewardAvailable = node.rewardAvailable.add(
        node.rewardAvailable.mul(_productionRatePer).div(
            PERCENT_DENOMINATOR
        )
    );
}
```

Security Threads

Owner can change user's node monthly fee tax, claim tax and isolation period.

```
function updateMonthlyFee(address _user, uint256 _nodeIndex, uint256 _updatedFee
) external isRole(NODE_BOOSTER_ROLE) {
    NodeEntity storage node = _nodesOfUser[_user][_nodeIndex];
    require(
        node.boosterInfo.royalFox < perNodeBoosterLimit,
        "NODE: Already Boosted"
    );
    node.boosterInfo.royalFox += 1;
    node.monthlyFee.fee = _updatedFee;
}
function updateNodeClaimTax(address _user, uint256 _nodeIndex, uint256 _taxPer
) external isRole(NODE_BOOSTER_ROLE) {
    NodeEntity storage node = _nodesOfUser[_user][_nodeIndex];
    require(
        node.boosterInfo.luckyYard < perNodeBoosterLimit,
        "NODE: Already Boosted"
    );
    node.boosterInfo.luckyYard += 1;
    node.claimTax = node.claimTax.sub(
        node.claimTax.mul(_taxPer).div(PERCENT_DENOMINATOR)
    );
}
function updateIsolationPeriod(address _user, uint256 _nodeIndex, uint256 _days
) external isRole(NODE_BOOSTER_ROLE) {
    NodeEntity storage node = _nodesOfUser[_user][_nodeIndex];
    require(
        node.boosterInfo.sparkTouch < perNodeBoosterLimit,
        "NODE: Already Boosted"
    );
    node.boosterInfo.sparkTouch += 1;
    node.isolationPeriod = node.isolationPeriod.sub(_days);
}
```

Contexia Project & Team Review

According to project’s whitepaper:
Contexia will be passive income protocol that combines Node with NFTs, gambling, and gamification. Contexia's nodes will be called Jackpot.
Jackpots come in three different collections: Silver Paltis, Diamond Lucius, and Golden Emis. Holder of jackpot nodes will earn $CON at a fixed rate as long as he owns the Jackpot.

The future development plans of Contexia are as follows:

- Launch casino
- Release Contexia DAO
- Release new NFTs

Team:

Team is KYCed by Assure Defi

https://www.assuredefi.io/projects/contexia-finance/

Website Analysis

URL: https://contexia-finance.com/

- **Design:** Single page, simple design and pleasant color scheme.
- **Content:** There is no content except socials and whitepaper.
- **Whitepaper:** Well written and explanatory, no grammar mistakes.
- **Roadmap:** Goals set at 4 phases without time frames.
- **Mobile-friendly?** Yes
- **Technical:** SSL certificate present. General SEO check passed.

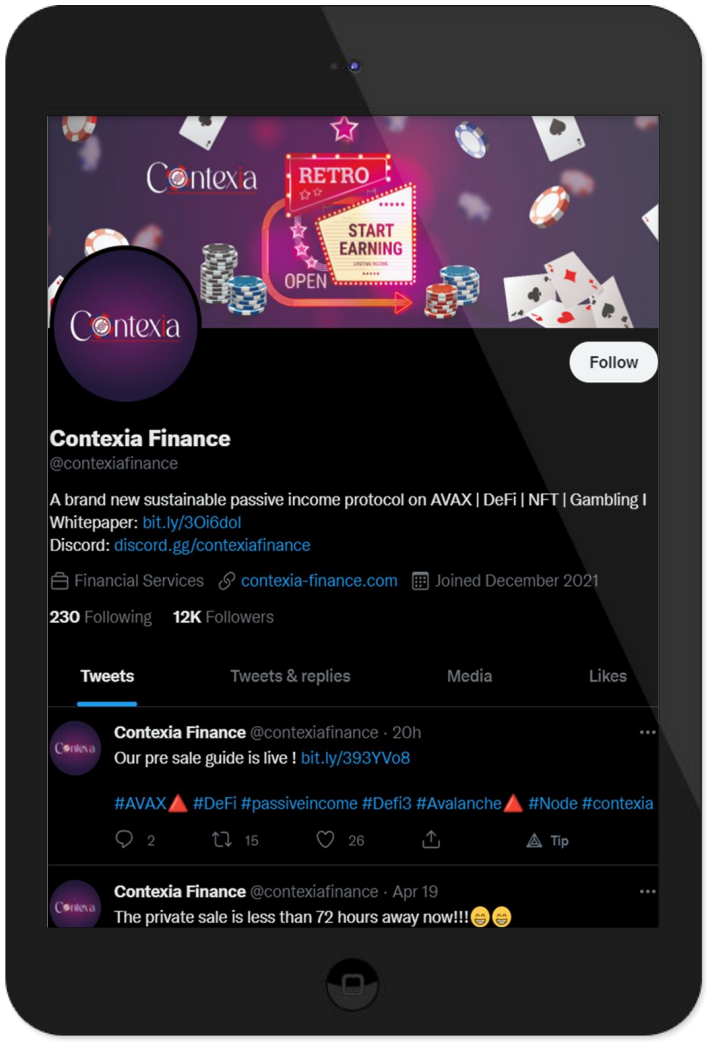
Social Media & Online Presence

Discord

https://discord.com/invite/contexiafinance

-5 541 members

- Active



Twitter

https://twitter.com/contexiafinance

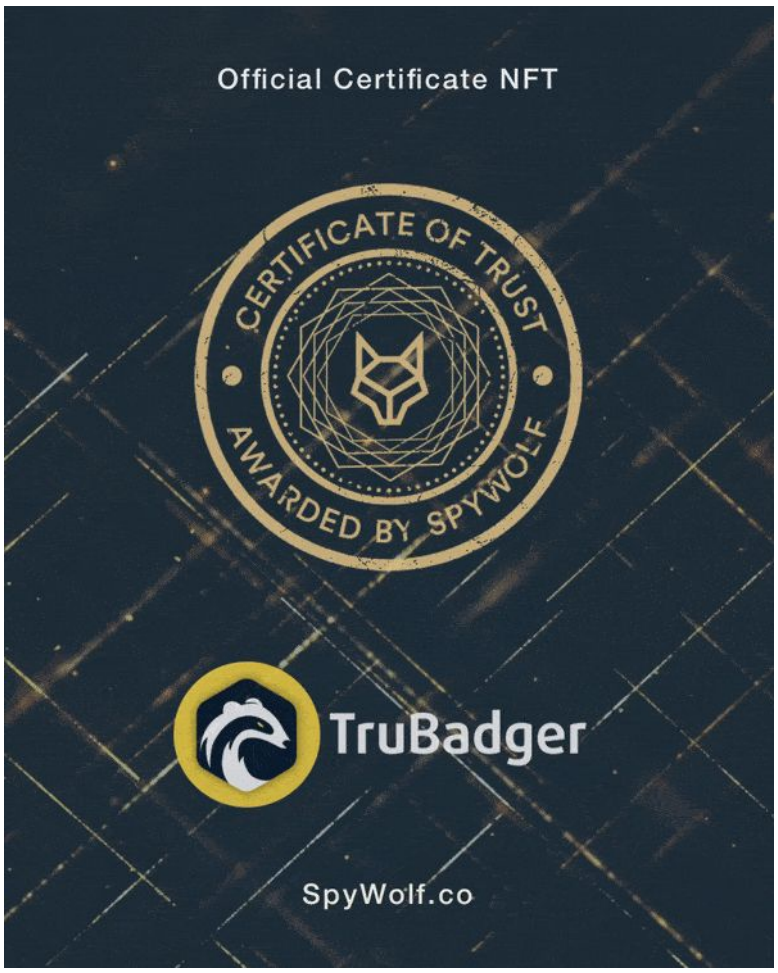
- 12 000 Followers

- Active

About SpyWolf

SpyWolf is a team of crypto security experts that have been performing full audits for projects for the past months in order to ensure safety on the crypto space. Our goal is to help eliminate monetary fraud through our auditing services and utility token, $SPY.

- Website: SpyWolf.co
- Portal: SpyWolf.network
- Telegram: @SpyWolfNework
- Twitter: Twitter.com/SpyWolfNetwork



(Sample Certificate NFT for those who pass audit)

If you are interested in finding out more about our audits and Certificate of Trust NFTs, reach out to contact@spywolf.co.

Disclaimer

This report shows findings based on our limited project analysis, following good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall social media and website presence and team transparency details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report.
While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER:
By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.
No one shall have any right to rely on the report or its contents, and SpyWolf and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (SpyWolf) owe no duty of care towards you or any other person, nor does SpyWolf make any warranty or representation to any person on the accuracy or completeness of the report.
The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and SpyWolf hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, SpyWolf hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against SpyWolf, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts, website, social media and team.
No applications were reviewed for security. No product code has been reviewed.